

# Adaptive finite element simulation of Stokes flow in porous media

J. Cao & P. K. Kitanidis

Department of Civil and Environmental Engineering, Stanford University, Stanford, CA 94305-4020, USA

(Received 18 April 1997; accepted 25 August 1997)

The Stokes problem describes flow of an incompressible constant-viscosity fluid when the Reynolds number is small so that inertial and transient-time effects are negligible. The numerical solution of the Stokes problem requires special care, since classical finite element discretization schemes, such as piecewise linear interpolation for both the velocity and the pressure, fail to perform. Even when an appropriate scheme is adopted, the grid must be selected so that the error is as small as possible. Much of the challenge in solving Stokes problems is how to account for complex geometry and to capture important features such as flow separation. This paper applies adaptive mesh techniques, using *a posteriori* error estimates, in the finite element solution of the Stokes equations that model flow at pore scales. Different selected numerical test cases associated with various porous geometries are presented and discussed to demonstrate the accuracy and efficiency of our methodology. © 1998 Elsevier Science Limited.

## 1 INTRODUCTION

In a hydrogeologic and environmental study, the flow of groundwater and the transport of solutes are represented at scales much larger than that of a pore unit. Usually, the medium is viewed as a continuum and Darcy's scale becomes the smallest scale of interest. Flow is governed by an equation based on Darcy's law and continuity of bulk mass, and the transport and fate of solutes is governed by the advection-dispersion-reaction equation with bulk fluxes.<sup>8,16</sup>

The study of flow and transport mechanisms at the pore scale is much more difficult but is inevitable for a better understanding of flow and transport in porous media. A case in point is the presence of zones of immobile water which may impose mass transfer limitations and thus affect reaction rates, advective mass transport rates, and dispersive mixing.<sup>14</sup> There is considerable interest in understanding how immobile regions may be created and how flow may be reserved in 'dead end' pores and in fissures. The investigation of such particular flow phenomena is a prerequisite for the study of heterogeneous reactions, such as in the presence of biofilms.<sup>12,13</sup>

The Navier–Stokes equations are considered as an ideal model in numerically simulating the flow of incompressible fluids with constant viscosity. However, the nonlinear

Navier–Stokes equations require a complicated and expensive solution process. Fortunately, a mathematically simplified model, which is called the Stokes problem, can be reduced from the Navier–Stokes equations physically in the case of slow flow, i.e. when the Reynolds number is small. In fact, low Reynolds numbers are the rule for porous medium flows.<sup>19,28</sup>

Numerical approximations (previous studies have involved Stokeslets,<sup>20</sup> used the lattice-gas method,<sup>17</sup> the finite difference scheme<sup>30</sup> and boundary element simulation<sup>28</sup>) have been widely used in the solution of the Stokes equations. In our study, the geometry of real porous media is very complex, e.g. the solid boundary is wavering, the throat parts located among particles are very narrow resulting in relatively high velocities, etc. In this case, an efficient and flexible numerical tool is required to account for all the important features of the pore geometry. It is well known that the finite element approximation can achieve this goal and we adopt it in our numerical implementation.

Properties of existence, uniqueness, and regularity of the finite element solution of the Stokes equations have been studied previously.<sup>25,26,23,31</sup> Algorithmic and numerical implementation issues have also been discussed.<sup>32,18,23,21,27</sup> However, classical finite element methods with a single pre-generated grid seem inadequate for obtaining a satisfactory

numerical solution in the case of complicated geometry. In order to get a detailed and precise representation of flow in porous media, we use in this study a mesh adaptive algorithm coupled with *a posteriori* error estimate analysis.

Historically, this mathematical approach was introduced by Babuška for more general problems;<sup>2</sup> in the domain of elliptic equations, some results of theoretical research on *a posteriori* error estimates have been presented by Bank.<sup>3</sup> Cao<sup>11</sup> has studied adaptive grids based on *a posteriori* error estimates for the Stokes problem in its most generalized form and we will apply these results in our work. In short, the idea is to estimate the error within each element by solving a local Neumann problem corresponding to the mathematical model. Through solving this local Neumann problem, we obtain the *a posteriori* error estimates, which can be used to form both local and global energy norms involving all the physical variables. This energy norm is ultimately used to approximately equi-distribute the error index over the mesh.

Guided by this mathematical criterion, we implement mesh adaptation through a ‘bisection refinement’ algorithm<sup>5</sup> that can ensure a quite continuous granularity on the adapted grid. In addition to refinement, we can move points on the mesh within a reasonably small neighborhood to further improve the mesh quality.<sup>6</sup> The mesh refinement and node-moving techniques complement each other, allowing for great flexibility to achieve the best mesh possible.

The rest of the paper is organized as follows. Section 2 focuses on the Stokes problem: we first describe this mathematical model and some specific stylized geometries encountered in porous medium flow studies, then we explain the renowned ‘mini-element’ formulation specifically applied to the solution of the Stokes problem, and we schematize a Stokes solver of UZAWA type. Section 3 gives the *a posteriori* error estimator framework corresponding to the Stokes problem. Section 4 outlines different mesh adaptation schemes. In Section 5, we present results for three numerical test cases associated with typical porous computational domains and discuss the resulting different solution behaviors. We finally make some concluding remarks and discuss extension of this work in Section 6.

## 2 THE STOKES PROBLEM

### 2.1 Mathematical modeling

For illustration, consider flow inside a unit with sinusoidal wall boundary (see Fig. 1) isolated from a long periodic porous tube. We will determine the periodic flow inside it. Here, this flow is generally governed by the following Stokes equations:

$$\begin{cases} -\mu \Delta \vec{u} + \vec{\nabla} p = \vec{0} & \text{in } \Omega \\ \vec{\nabla} \cdot \vec{u} = 0 & \text{in } \Omega \end{cases} \quad (1)$$

where  $\vec{u}, p$  represent the velocity and the pressure, respectively; the positive parameter  $\mu$  is the dynamic viscosity of the fluid.

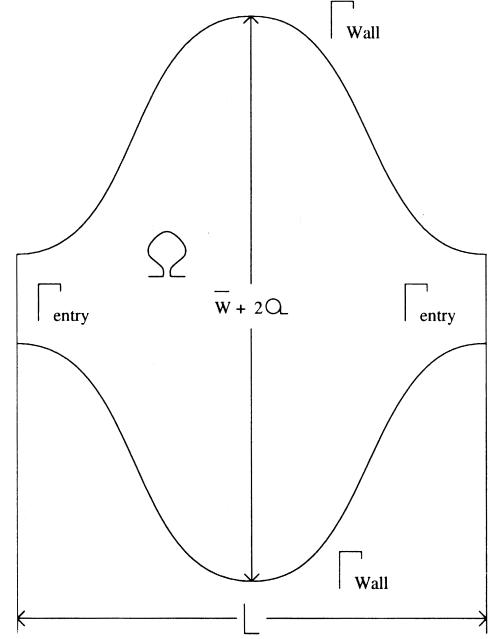


Fig. 1. Computational domain No. 1.

Boundary conditions need to be added as follows. For example, in the case of the pore shown in Fig. 1, we use the no-slip/no-flux condition for the solid surfaces of the channel:

$$\vec{u} = \vec{0} \text{ on } \Gamma_{\text{wall}} \quad (2)$$

and the conditions at the entrance and the exit are typically imposed by a certain parabolic  $\vec{u}$ -profile. Previously,<sup>22</sup> an approximate analytical approach was used to find the solution in a long sinusoidal channel. Consistent with that work, we set:

$$\vec{u} = \text{periodic velocity profile on } \Gamma_{\text{entry}}; \Delta p = \text{constant} \quad (3)$$

where  $\Delta p$  represents the reduction of pressure at two equivalent points on  $\Gamma_{\text{entry}}$ . In our numerical simulation, the results are presented assuming that the average pressure over the computational domain is zero.

More precisely, we enforce the periodic boundary condition by solving a sequence of problems. Starting with an initial velocity profile on  $\Gamma_{\text{entry}}$ , we solve the Stokes problem in a 2-unit domain shown in Fig. 2. The behavior of the velocity occurring at the middle throat  $\Gamma_{\text{entry}^*}$  is then collected; next, we combine the pre-imposed  $\vec{u}$ -profile at the starting and ending sections and the newly-collected  $\vec{u}$ -profile at the middle throat, this updated  $\vec{u}$ -profile serving as new Dirichlet boundary condition for the entrance and the exit. Such a loop converges rapidly to an identical  $\vec{u}$ -profile at all three throats.

As a complement, another numerical test case will be implemented in a domain with abruptly varying boundary so that we can observe some singular phenomena within the

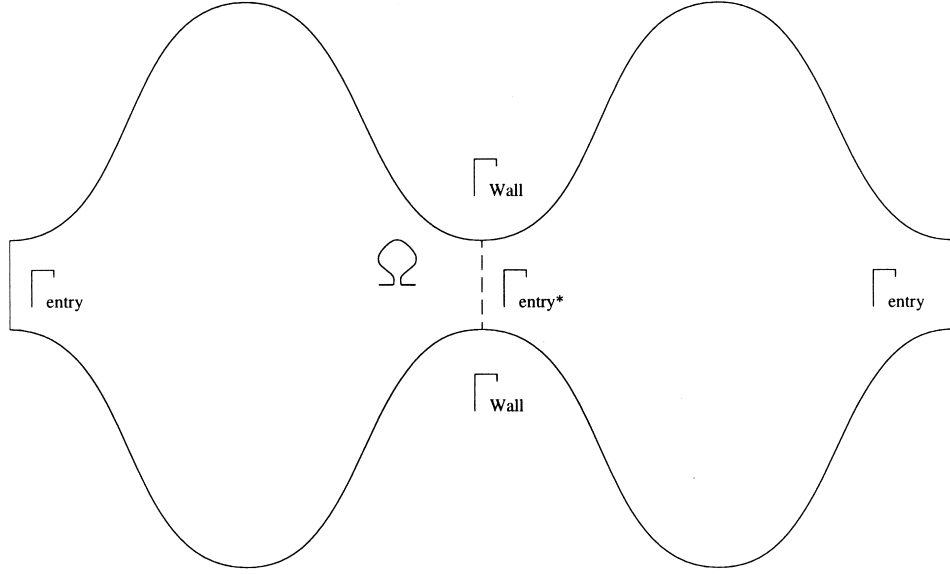


Fig. 2. Computational domain No. 2.

Stokes flow context. Such a step-like geometry is illustrated in Fig. 3.

## 2.2 Finite element discretization

To establish notation, let us review some functional and discrete spaces. For simplicity, we suppose that  $\Omega$  is a polygonal two-dimensional domain. The space  $L^2 = L^2(\Omega)$  is a regular set of square integrable functions over  $\Omega$ , on which  $(\cdot, \cdot)$  represents the usual inner product, associated with the norm  $\|\cdot\|$ . The space  $H^1 = H^1(\Omega)$  denotes the usual Sobolev space of functions in  $L^2$  whose first derivatives are also contained in  $L^2$ . Let  $C^0$  be the space of functions continuous on  $\Omega$ .  $\Omega_h$  represents here a standard finite element triangulation of  $\Omega$  with  $h$ , where the quantity  $h = \max_{\tau \in \Omega_h} h_\tau$  is a measure of the granularity of the triangulation. The spaces of linear and cubic polynomials are denoted as  $P_1$  and  $P_3$ , respectively. Now we can define the following discrete spaces:

$$H_h^1 = \{q_h | q_h \in C^0(\Omega), q_h|_\tau \in P_1, \forall \tau \in \Omega_h\} \quad (4)$$

$$V_{h_g}^1 = \{\vec{v}_h | \vec{v}_h \in (C^0(\Omega))^2, \vec{v}_h|_\tau \in P_{1\tau}^* \times P_{1\tau}^*, \vec{v}_h|_\Gamma = \vec{g},$$

$$\forall \tau \in \Omega_h\} \quad (5)$$

with

$$P_{1\tau}^* = \{q | q = q_1 + \kappa \phi_\tau, q_1 \in P_1, \kappa \in \mathbf{R}, \phi_\tau \in P_3,$$

$$\phi_\tau|_{\partial\tau} = 0, \phi_\tau(G_\tau) = 1\} \quad (6)$$

where  $G_\tau$  is the centroid of the triangle  $\tau$ . A function like  $\phi_\tau$  is usually called a bubble-function.<sup>1</sup>

In eqn (1), due to the incompressibility condition  $\vec{\nabla} \cdot \vec{u} = 0$ , the finite element discretization spaces for the velocity and the pressure need to satisfy a compatibility condition, also called ‘inf-sup condition’ or ‘Ladyzhenskaya–Babuška–Brezzi (LBB) condition’,<sup>9</sup> which is equivalent to the requirement of nonsingularity of the matrix resulting from the discretized Stokes system, and thus guarantees, if satisfied, the existence and uniqueness of the solution. In particular, this condition implies a higher number of velocity degrees of freedom than pressure unknowns. This condition

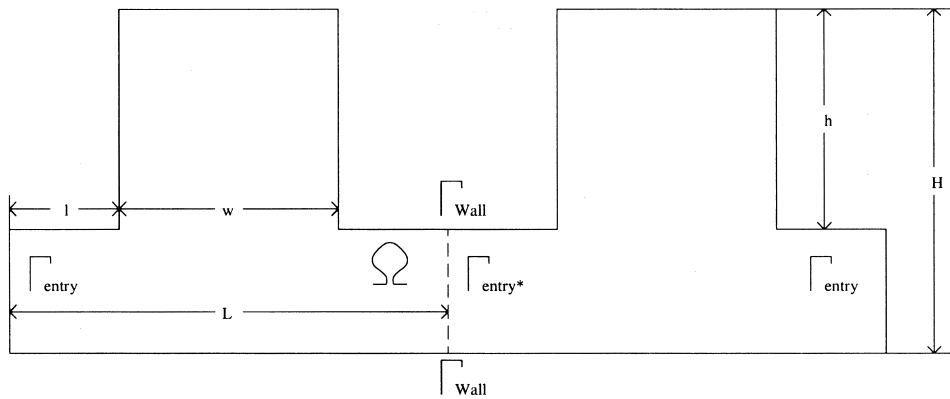


Fig. 3. Computational domain No. 3.

is:

$$\inf_{\substack{p \in L^2(\Omega) \\ p \neq 0}} \sup_{\substack{\vec{u} \in (H^1(\Omega))^2 \\ \vec{u} \neq 0}} \frac{(\vec{\nabla} \cdot \vec{u}, p)}{\|\vec{\nabla} \vec{u}\| \|p\|} \geq c > 0 \quad (7)$$

where  $c$  is a positive constant. Among possible space discretizations satisfying the LBB condition, we choose the ‘mini-element’ formulation,<sup>1</sup> which permits the resolution of the Stokes problem on a single grid for all the unknowns.

The basic idea of the mini-element formulation is that the pressure is discretized by polynomials of degree 1 ( $P_1$ ), while the velocity is also discretized by polynomials of degree 1, augmented by a polynomial of degree 3 (bubble-function) which vanishes on the edges of the triangle  $\tau$ . With the mini-element formulation, the discretized Stokes system is written as:

$$\begin{cases} \text{Find } (\vec{u}_h, p_h) \in V_{h_g}^1 \times H_h^1 \text{ such that} \\ \mu(\vec{\nabla} \vec{u}_h, \vec{\nabla} \vec{v}) - (p_h, \vec{\nabla} \cdot \vec{v}) = \vec{0} & \forall \vec{v} \in V_{h_0}^1 \\ -(q, \vec{\nabla} \cdot \vec{u}_h) = 0 & \forall q \in H_h^1 \end{cases} \quad (8)$$

where  $\vec{v}$  and  $q$  are the test functions associated with  $\vec{u}_h$  and  $p_h$ , respectively. Using elementwise integration, the ‘mini-element’ discretization leads eqn (8) to the following stiffness matrix equation of  $(\vec{u}_h \in V_{h_g}^1, p_h \in H_h^1)$  for each  $\tau$ :

$$\begin{pmatrix} A_\tau & e_\tau & 0 & 0 & B_{\tau,x}^t \\ e_\tau' & \sigma_\tau' & 0 & 0 & w_{\tau,x}^t \\ 0 & 0 & A_\tau & e_\tau & B_{\tau,y}^t \\ 0 & 0 & e_\tau' & \sigma_\tau' & w_{\tau,y}^t \\ B_{\tau,x} & w_{\tau,x} & B_{\tau,y} & w_{\tau,y} & 0 \end{pmatrix} \begin{pmatrix} u_{h,l} \\ u_{h,b} \\ v_{h,l} \\ v_{h,b} \\ p_{h,l} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad (9)$$

where the solution  $(\vec{u}_h, p_h)$  can be uniquely decomposed into its linear part  $(\vec{u}_{h,l}, p_h)$  at three vertices of the triangle  $\tau$  and its bubble part  $(\vec{u}_{h,b}, 0)$  at the centroid of  $\tau$  (here, the subscripts  $l$  and  $b$  represent the piecewise linear and cubic interpolations, respectively); thus, for the triangle  $\tau$ , there are totally 11 unknowns: 4 unknowns for each component of the velocity and 3 for the pressure. The  $3 \times 3$  matrices  $A_\tau, B_{\tau,x}, B_{\tau,y}$  correspond to inner products involving linear basis function for the velocity and the pressure, the scalar  $\sigma_\tau'$  is the contribution to the  $H^1$  inner product from the cubic bubble functions for the velocity, and the 3-vectors  $w_{\tau,w}$  and  $w_{\tau,y}$  correspond to contributions to the divergence term for bilinear basis functions (1-order derivative of the cubic bubble functions for the velocity) and linear functions (for the pressure). The detailed expressions of each bloc in eqn (9) can be found elsewhere<sup>11</sup> where the Stokes problem is studied in its most general form. For further discussion, we specify here only the expression of  $\sigma_\tau'$ :

$$\sigma_\tau' = \frac{\mu(h_1^2 + h_2^2 + h_3^2)}{720|\tau|} \quad (10)$$

where  $h_i$  ( $i = 1, 2, 3$ ) and  $|\tau|$  are the length of three edges and the area of  $\tau$ , respectively.

As mentioned, the bubble function turns out to be zero on the triangle edges. We use in fact only the linear part of the velocity solution. In practice, we can eliminate formally the bubble unknowns  $u_{h,b}$  and  $v_{h,b}$  from the left-hand side of eqn (9). This can be done elementwisely. With the bubble unknowns statically condensed,<sup>11</sup> eqn (9) is now reduced into the following form:

$$\begin{pmatrix} A_\tau' & 0 & B_{\tau,x}^t \\ 0 & A_\tau' & B_{\tau,y}^t \\ B_{\tau,x}^t & B_{\tau,y}^t & -C_\tau' \end{pmatrix} \begin{pmatrix} u_{h,l} \\ v_{h,l} \\ p_{h,l} \end{pmatrix} = \begin{pmatrix} F_1 \\ F_2 \\ H \end{pmatrix} \quad (11)$$

Interested readers are referred elsewhere<sup>11</sup> for the details of each bloc in eqn (11). We give here only the form of  $C_\tau'$  which is no longer zero as in eqn (9):

$$C_\tau' = \frac{1}{\sigma_\tau} (w_{\tau,x} w_{\tau,x}^t + w_{\tau,y} w_{\tau,y}^t) \quad (12)$$

In fact, the solvability of the Stokes problem is owing to the condensation terms appearing in eqn (12).

### 2.3 A Stokes solver

The assembly of all elementwise contributions and right-hand sides leads the Stokes system of eqn (11) to the following global equation:

$$\begin{pmatrix} \bar{A} & \bar{B} \\ \bar{B}^t & \bar{C} \end{pmatrix} \begin{pmatrix} \bar{u} \\ \bar{p} \end{pmatrix} = \begin{pmatrix} \bar{F} \\ \bar{H} \end{pmatrix} \quad (13)$$

We solve the linear Stoke problem using a multigrid algorithm with hierarchical basis.<sup>4</sup> Since we will adopt in this study the ‘longest edge bisection’ rule for triangle refinement (see Section 4 for details) whose hierarchical nature is not obvious, an auxiliary process is required to help us construct an ‘artificial multigrid’ structure on an arbitrary mesh according to the distribution of granularity of the triangulation.<sup>7</sup> With transitions within V-cycle multigrid structure, the following UZAWA-like conjugate gradient method<sup>15,24,10</sup> is used to solve the whole Stokes system.<sup>11</sup>

#### Step 1:

give  $\epsilon \ll 1$ ;  
give  $\bar{p}^0$ ;  
solve  $\bar{A}\bar{u}^0 = \bar{F} - \bar{B}\bar{p}^0$ ;  
 $g^0 = \bar{H} - \bar{B}^t\bar{u}^0 - \bar{C}\bar{p}^0$ ;  
if  $\|g^0\| \leq \epsilon$ , the solution  $\bar{u} = \bar{u}^0$ ,  $\bar{p} = \bar{p}^0$  is achieved;  
otherwise go to Step 2.

#### Step 2:

solve  $S\bar{g}^0 = g^0$ ;  
 $h^0 = -\bar{g}^0$ ;  
solve  $\bar{A}\chi^0 = -\bar{B}h^0$ ;  
 $\eta^0 = -\bar{C}h^0 - \bar{B}^t\chi^0$ ;  
 $\alpha^0 = (g^0, \bar{g}^0)/(\eta^0, h^0)$ ;

$\bar{u}^1 = \bar{u}^0 + \alpha^0 \chi^0$ ;  
 $\bar{p}^1 = \bar{p}^0 + \alpha^0 h^0$ ;  
 $g^1 = g^0 + \alpha^0 \eta^0$ ;  
 if  $\|g^1\| \leq \epsilon$ , the solution  $\bar{u} = \bar{u}^1$ ,  $\bar{p} = \bar{p}^1$  is achieved;  
 otherwise go to Step 3.

**Step 3:** (for  $m \geq 1$ ):

solve  $S\bar{g}^m = g^m$ ;  
 $\gamma^m = (g^m, \bar{g}^m)/(g^{m-1}, \bar{g}^{m-1})$ ;  
 $h^m = -\bar{g}^m + \gamma^m h^{m-1}$ ;  
 solve  $\bar{A}\chi^m = -\bar{B}h^m$ ;  
 $\eta^m = -\bar{C}h^m - \bar{B}^t \chi^m$ ;  
 $\alpha^m = (g^m, \bar{g}^m)/(\eta^m, h^m)$ ;  
 $\bar{u}^{m+1} = \bar{u}^m + \alpha^m \chi^m$ ;  
 $\bar{p}^{m+1} = \bar{p}^m + \alpha^m h^m$ ;  
 $g^{m+1} = g^m + \alpha^m \eta^m$ ;  
 if  $\|g^{m+1}\| \leq \epsilon$ , the solution  $\bar{u} = \bar{u}^{m+1}$ ,  $\bar{p} = \bar{p}^m$  is achieved; otherwise replace  $m$  by  $m + 1$  and go to the beginning of Step 3.

In Steps 2 and 3,  $S$  is a preconditioner and this preconditioning is optional.

### 3 A POSTERIORI ERROR ESTIMATOR

We have chosen an *a posteriori* error estimate as the criterion of our adaptive mesh algorithms. The complete derivation and detailed analysis of our estimator is too lengthy to be presented in this paper. The interested reader is referred elsewhere<sup>11</sup> for more details on the *a posteriori* error estimate for the generalized Stokes problem. Here, we just reduce that research result into the case of the Stokes problem (eqn (1)).

Our *a posteriori* error estimator requires the element-by-element solution of a local Neumann problem, discretized by a finite element discretization of higher order, to obtain a local error indicator for each physical variable  $\bar{u}$  and  $p$  in the Stokes problem as expressed in eqn (1). Let  $B_\tau$  be the set of quadratic polynomials over  $\tau$  which are zero at the vertices of  $\tau$ . Because in Section 2 we solve the Stokes problem using linear elements (even with the mini-element formulation, the velocity is represented by only its linear part after the static condensation of bubble unknowns), the local error approximations should be computed with at least the ‘bump’ basis functions constructed in  $B_\tau$ . Then,  $(\bar{e}_S, \epsilon_S) \in (B_\tau)^3$  will be elementwisely solved, the error unknowns arising at the middle points of the three edges of the triangle  $\tau$ ; for the solution group  $(\bar{u}, p)$  each has 3 error unknowns on three edges, totally resulting in 9 error unknowns. The local approximate error estimate  $(\bar{e}_S, \epsilon_S) = (\bar{u}, p) - (\bar{u}_{h,l}, p_{h,l})$  is computed by solving the following  $9 \times 9$  linear equation for each triangle  $\tau$ :

with  $(\bar{u}_{h,l}, p_h)$ , the piecewise linear solution resulting from the mini-element formulation (for the velocity, only the linear part of the numerical solution is used). In eqn (14), the notation of  $(\cdot, \cdot)_\tau$  represents the usual inner product strictly over triangle  $\tau$  while  $\langle \cdot, \cdot \rangle_{\partial\tau}$  represents the inner product along the boundary of  $\tau$ ;  $[\partial\bar{u}_{h,l}/\partial n]_A$  is the average normal derivative of  $\bar{u}_{h,l}$  across the edges of  $\tau$ ;  $\lambda_\tau$  a positive multiple for each triangle  $\tau$ , we define it as:

$$\lambda_\tau = \frac{|\tau|}{3600\sigma_\tau} \quad (15)$$

with  $\sigma_\tau$  given in eqn (10).

Finally, using these local error indicators, we can form both local and global energy norms:

$$\|(\bar{e}, \epsilon)\|^2 = \mu \|\bar{\nabla} \bar{e}\|^2 + \frac{1}{\mu} \|\epsilon\|^2 \quad (16)$$

for use in our adaptive mesh algorithms.

### 4 ADAPTIVE MESH ALGORITHMS

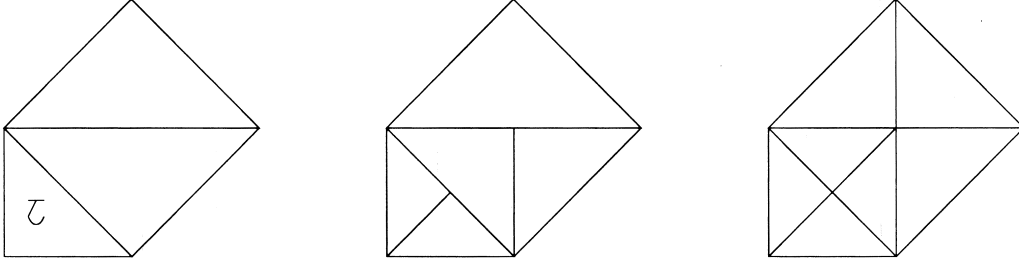
A simple bisection algorithm,<sup>5</sup> called longest edge bisection, is applied in this work. According to the suggestion of Rivara,<sup>29</sup> elements selected for refinement are bisected along their longest edge. The neighbor element sharing the longest edge is also bisected along its longest edge. If the result is a compatible triangulation, the process stops. Otherwise, it is recursively applied to the longest edge neighbors of all refined elements. An example is shown in Fig. 4. This process is known to have finite termination, typically in a very small number of steps.

Although there is also a natural tree data structure which could be used in this case, we have implemented our longest edge bisection algorithm with no refinement tree, keeping only those elements which are currently in the mesh. This has allowed us to study coarsening algorithms for use on completely unstructured meshes. Such algorithms have applications in the adaptive unrefinement of an unstructured mesh, as well as certain algebraic hierarchical basis multi-level iterative methods.<sup>7</sup>

For both refinement and unrefinement algorithms, *a posteriori* error estimates are used to decide which elements to refine/unrefine. The guiding principle is that of mesh equilibration; i.e. we attempt through the refinement/unrefinement process to create a final mesh in which all elements have approximately the same error regardless of size.

Our mesh moving algorithm also uses an *a posteriori* error estimator, but in a slightly different fashion. In our algorithm, the mesh topology (connectivity) remains fixed, but the locations of the mesh points themselves are allowed to move in response to the error estimator. Our procedure

$$\begin{cases} \mu(\bar{\nabla} \bar{e}_S, \bar{\nabla} \bar{v})_\tau - (\epsilon_S, \bar{\nabla} \cdot \bar{v})_\tau = -\mu(\bar{\nabla} \bar{u}_{h,l}, \bar{\nabla} \bar{v})_\tau + (p_h, \bar{\nabla} \cdot \bar{v})_\tau + \langle \mu \left[ \frac{\partial \bar{u}_{h,l}}{\partial n} \right]_A - p_h \bar{n}, \bar{v} \rangle_{\partial\tau} & \forall \bar{v} \in B_\tau \times B_\tau \\ -(q, \bar{\nabla} \cdot \bar{e}_S)_\tau - \lambda_\tau(\bar{\nabla} \epsilon_S, \bar{\nabla} q)_\tau = (q, \bar{\nabla} \cdot \bar{u}_{h,l})_\tau + \lambda_\tau(\bar{\nabla} p_h, \bar{\nabla} q)_\tau & \forall q \in B_\tau \end{cases} \quad (14)$$

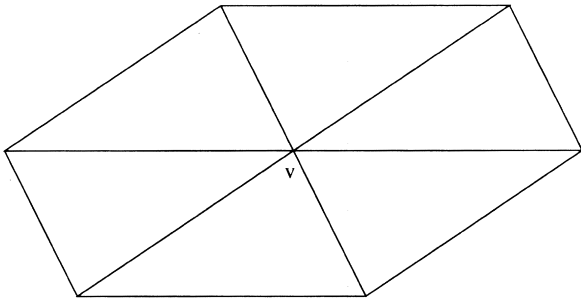


**Fig. 4.** Element  $\tau$  is refined by the longest edge bisection method: from the original mesh (left), the first step of bisection (middle) does not yet yield a compatible triangulation; however, the second step (right) does yield a triangulation.

consists of a Gauss–Seidel-like iteration on the vertices in the mesh, in which the position of each vertex is locally optimized with all other vertices held fixed. This is illustrated in Fig. 5, where the position of vertex  $v$  is optimized within the region  $\Omega_v$  by approximately minimizing the *a posteriori* error estimate with respect to vertex location. Each optimization problem has two degrees of freedom, and is solved using a simple approximate Newton iteration. Details of this algorithm can be found elsewhere.<sup>6</sup> Not all vertices in the mesh are allowed to move. Some boundary and interface vertices must remain fixed to preserve the definition of the region; others are allowed only one degree of freedom.

## 5 NUMERICAL EXPERIMENTS

We will present in this section three typical test cases of Stokes flow simulation. The first two are associated with sinusoidal channels. For this geometry, with  $\bar{w}$ ,  $\bar{w} + 2a$ , and  $L$  denoting the pore unit’s average width, maximum width, and length, respectively (see previous work<sup>14</sup> for specific details on geometry definition),  $a/\bar{w}$  and  $L/\bar{w}$  are the two most important parameters. The former describes the narrowness of throat parts in the tube; the latter describes how gradually the width changes. The computation will be implemented on single-pore domain (see Fig. 1) and on double-pore domain (see Fig. 2), respectively, verifying the reasonableness of Dirichlet boundary condition setting presented in Section 2.1. The third numerical experiment will focus on flow simulation in steeply changing step-like



**Fig. 5.** The subregion  $\Omega_v$ , associated with vertex  $v$ .

channel (see Fig. 3) as we are particularly interested in flow behavior in the case of large fluctuations.

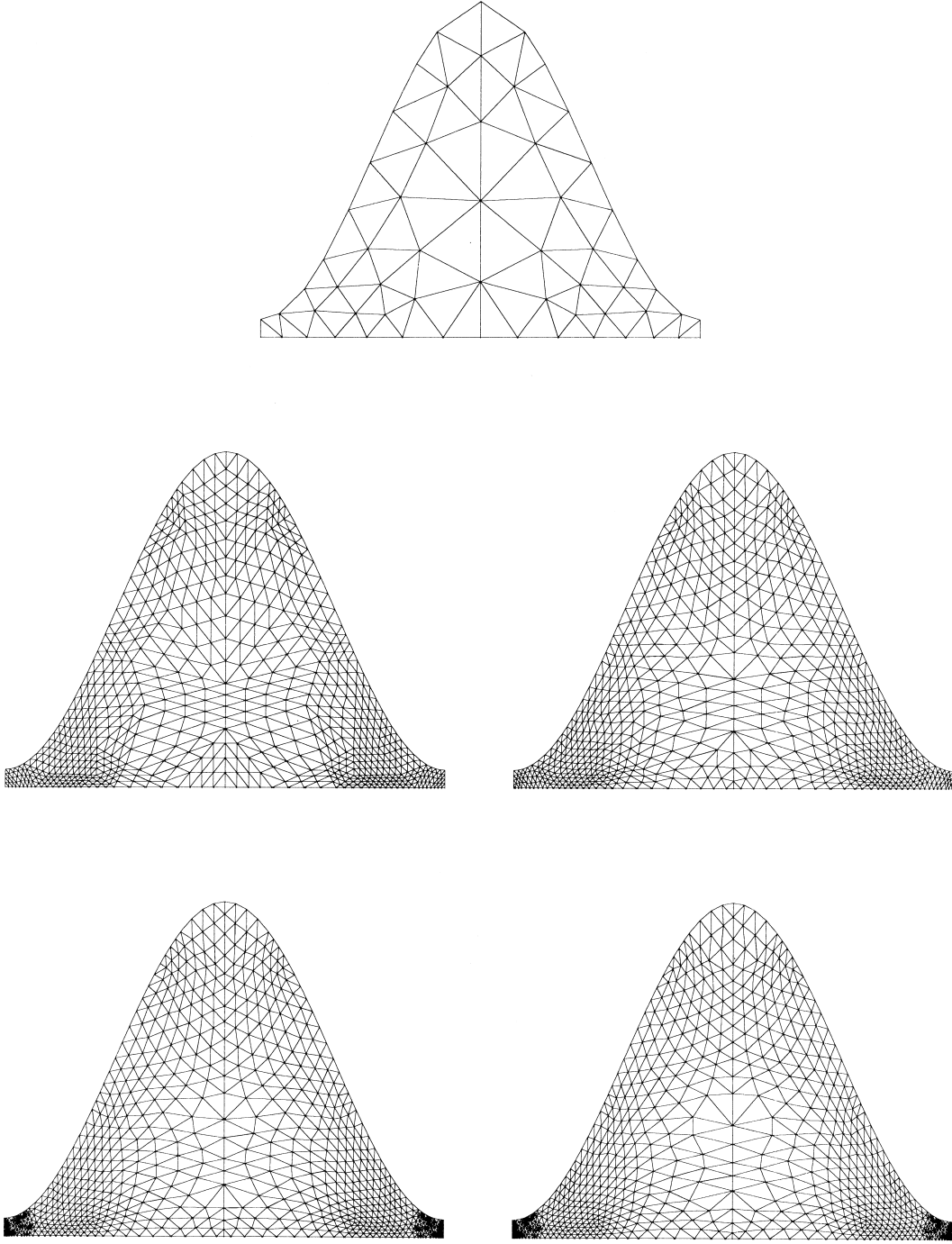
### 5.1 Test case 1: flow in single pore with $a/\bar{w} = 0.45$ , $L/\bar{w} = 1.25$

Our first test case is to simulate the Stokes flow in a single pore unit which looks rather sharp. The finite element mesh is initialized with 105 nodes and 168 triangles (see the upper part of Fig. 6, only the upper half of the whole grid is shown in this set of figures).

Based on the *a posteriori* error estimate analysis, the points on the grid are then slightly moved, and we further subdivide each triangle into 16 to get enough points, which yields an intermediate mesh ( $NV = 1425$ ,  $NT = 2688$ ) with a quite obvious hierarchical refinement structure (see the middle-left grid in Fig. 6). After moving nodes, the marks of the nested refinement disappear, and the density of points on the resulting mesh becomes much more continuous (see the middle-right grid in Fig. 6).

Then, the mesh is adapted to the flow behavior by several sweeps of refinement towards the point target number  $NV_{\text{target}} = 2000$  and, finally, we obtain a grid with 1999 nodes and 3784 elements (see the lower-left grid in Fig. 6). As the figure clearly demonstrates, the throat areas are remarkably refined due to the fastest flow velocity arising there; additionally, the imposition of non-homogeneous Dirichlet boundary condition can be described accurately only through a large number of points. Corresponding to this grid, the streamline distribution is demonstrated in the upper part of Fig. 7. The flow through the upper half of the pore unit is viewed as 1 here. Thus, from the left plotting, we see that the streamline corresponding to 0-streamfunction-value separates the main-flow zone from a backwater zone. In the main-flow area, 1/10 of the discharge passes between two consecutive plotted streamlines; it is obvious that most of the flow takes place near the center of the pore. A more detailed picture on the flow within the backwater zone is shown in the right plot in which each two neighboring streamlines occupies only 0.1% of the discharge, which means that the eddy taking place in the backwater zone is very weak.

Finally, we move nodes once more to reduce errors, yielding a mesh demonstrated in the lower-right corner of



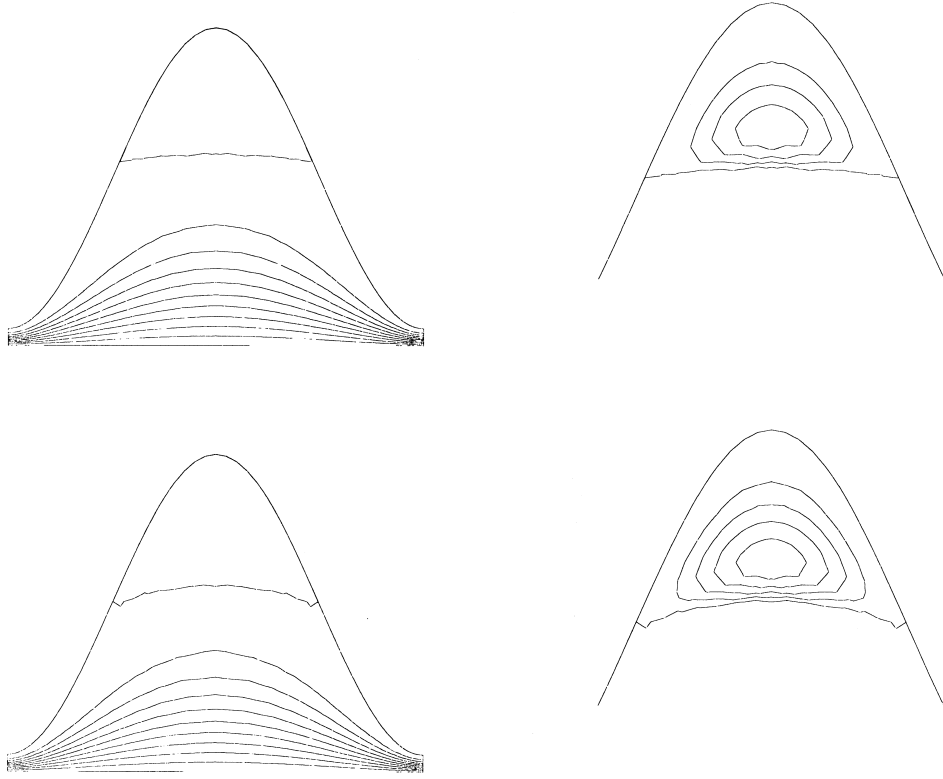
**Fig. 6.** Case 1: mesh evolution (upper: initial mesh; middle: intermediate meshes; lower: final meshes).

Fig. 6. The corresponding numerical streamline solution is depicted in the lower part of Fig. 7. We keep here the same scale used for the previous streamline plotting, and notice that the most important difference is the change of the 0-streamfunction-value line. After moving points, this line turns out perpendicular to the solid wall, which reflects more closely the physical situation.

We plot in Fig. 8 the velocity and pressure solution associated with the final mesh. We remark that in our work, the

maximum horizontal velocity component is normalized to 1, and all velocity illustrations in this paper are subject to this rule. The symmetry of iso-solution-value-line distributions implies to some degree the periodicity of the Stokes flow in the tube. In addition, the rather small negative horizontal velocity component ( $u_{\min} = -7.41e-4$ ) confirms again that there exists a very weak vertex in the pore.

Complete CPU time statistics for this test case are displayed in Table 1. We note that several sweeps of moving



**Fig. 7.** Case 1: streamline on final meshes (upper: before moving nodes; lower: after moving nodes).

grid nodes are imposed following mesh refinements in order to improve the quality of the resulting mesh. This makes the implementation of mesh refinement use more CPU time. Overall, 50% of the total CPU time is devoted to mesh adaptation, the other half used for the solution of the Stokes problem on successive meshes.

## 5.2 Test case 2: flow in double pores with $a/\bar{w} = 0.45$ , $L/\bar{w} = 2.5$

Next we double the unit length compared to the previous test case to make the pore flatter. We set two units as computational domain so that the velocity profile to be imposed on the entrance and exit sections can be modified with the numerical solution at the middle neck until they converge to the same profile. With the aid of *a posteriori* error estimate, we continue to use here exactly the same mesh adaptive process as in the previous implementation, involving

uniformly subdividing elements, adaptive refinements, and moving points.

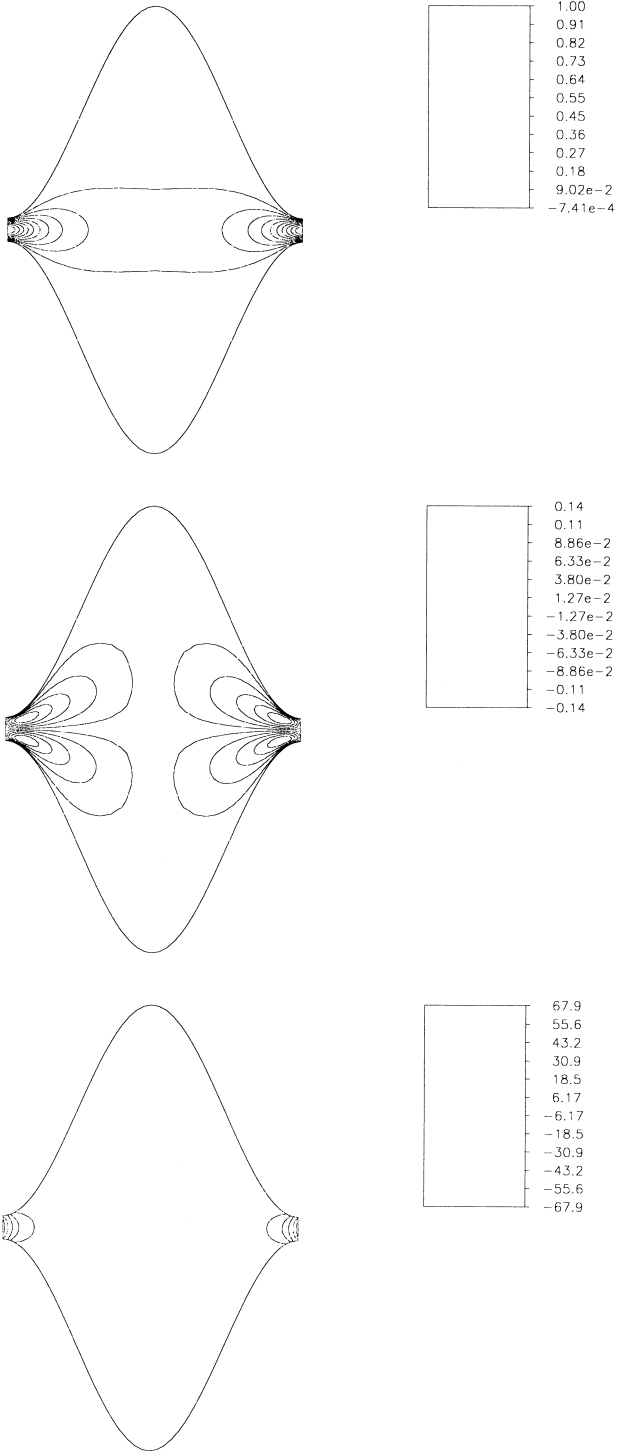
Fig. 9 illustrates the mesh evolution in this computation (we demonstrate only a quarter of meshes in this set): the upper grid is initially generated with 115 points and 160 elements; the intermediate mesh ( $NV = 1417$ ,  $NT = 2560$ ) is located in the middle; the lower grid with 1997 nodes and 3664 triangles is finally obtained via mesh adaptations. As in the previous case, the most sensitive area is located at necks where most grid refinement takes place.

Associated with the final mesh, the velocity and pressure solutions are shown in Fig. 10. Since the channel here is changing very slowly with respect to the previous case, we notice that the horizontal velocity component remains positive all the way. This leads to no backwater area in the channel seen from the streamline distribution in Fig. 11. Let us focus on the middle throat area, where nodes are quite dense following grid refinements. Using

**Table 1.** CPU time statistic for Case 1

Mesh adaptation description	Error estimate CPU	Adaptation CPU	Solution CPU	Total CPU
Moving nodes	0.05	0.09	0.20	0.34
Subdivision $NV$ : 105 $\rightarrow$ 1425	0.05	0.05	3.47	3.57
Moving nodes	0.73	1.98	3.20	5.91
Refinement $NV$ : 1425 $\rightarrow$ 1777	0.72	6.31	3.77	10.80
Refinement $NV$ : 1777 $\rightarrow$ 1999	0.88	7.33	3.94	12.15
Moving nodes	0.98	2.88	4.67	8.53





**Fig. 8.** Case 1:  $u$  (upper),  $v$  (middle), and  $p$  (lower) solutions on final mesh (after moving nodes).

the analytical solution studied previously<sup>22</sup> as reference, we find in the comparison shown in Fig. 11 that the streamline behavior obtained by our adaptive finite element method shares a practically identical distribution with the available analytical results. The periodicity of the flow appears clearly for each of the four physical variables: two velocity

components, pressure, and streamline, depicted in Figs 10 and 11, respectively.

Finally, we take a brief look at the error behavior. The indices of errors concerning the velocity part ( $\ln(\|\vec{\nabla}\tilde{e}\|/\|\vec{\nabla}\tilde{u}\|)$ ) and the density part ( $\ln(\|\epsilon\|/\|p\|)$ ) are respectively illustrated in Fig. 12, each two consecutive columns representing one mesh adaptive implementation. With five successive mesh adaptations, the errors decrease gradually.

### 5.3 Test case 3: flow in double step-like pores with $l/L = 0.3$ , $w/L = 0.4$ , $h/L = 0.8$ , and $H/L = 1.2$

Finally, we simulate Stokes flow in a step-like pore structure. The relevant geometric parameters are defined in Fig. 3. Unlike in the previous two cases, there exist singular points on the solid boundary, associated with corners on the wall.

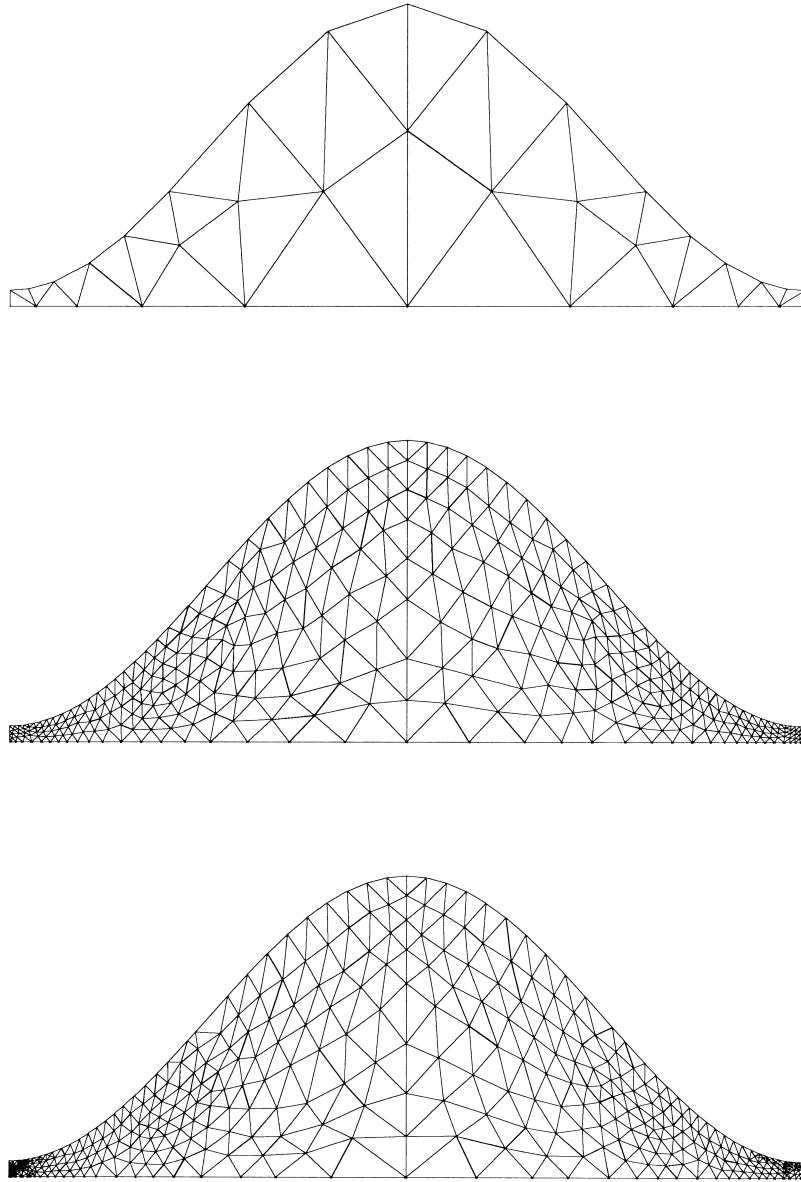
The initial triangulation shown in Fig. 13 is quite coarse, containing only 55 nodes and 68 triangles. An intermediate mesh is produced following uniform subdivision from one element into 49 and some auxiliary node-movings. The final grid towards  $NV_{\text{target}} = 2000$  is obtained through a further adaptive refinement followed by moving nodes once more. The intermediate and final meshes are shown in Fig. 13. Corresponding to this mesh set, we plot also a reference grid with a bigger size ( $NV = 2251$ ,  $NT = 4212$ ) than the final adapted mesh, and a comparison of the error behavior will be implemented on these two grids.

In this adaptive mesh evolution, the refinement occurs mainly around singular points (i.e. corners where velocity is theoretically unbounded) where the biggest error is located according to the result of *a posteriori* error estimate computation. The opposite happens at cornes which are stagnation points: flow rates are low, computed errors are small, and thus few points are needed.

Associated with the final mesh, the velocity and pressure solutions of the Stokes problem are demonstrated in Fig. 14. The periodicity is still clearly shown on these iso-solution-value-line distributions. Compared to the first test case, the negative horizontal velocity component appears much stronger ( $u_{\min} = -1.39e-2$  versus  $u_{\min} = -7.41e-4$  in the first case). This means the fluid rotates in the hollow zone more intensely in this case than in the first case.

Fig. 15 demonstrates the streamline behavior in this step-like channel. The left plot divides the total flow into 10 parts, the top two lines correspond to the same 0-streamfunction value, which encloses a vortex inside. The right plot details the vortex behavior within the backwater zone surrounded by the 0-streamfunction-value line, 2.5% (as opposed to 0.1% in the first test case) of the discharge passing between two consecutive plotted streamlines.

We finally remark that the resulting error estimates on the final adapted mesh are  $\mu\|\vec{\nabla}\tilde{e}\|^2 = 0.264$  and  $(1/\mu)\|\epsilon\|^2 = 0.0395$  in connection with the velocity and the pressure parts, respectively, while the reference mesh with a globally uniform point-distribution yields  $\mu\|\vec{\nabla}\tilde{e}\|^2 = 0.300$  and  $(1/\mu)\|\epsilon\|^2 = 0.0521$ , though the latter has more nodes than the former.



**Fig. 9.** Case 2: mesh evolution (upper: initial mesh; middle: intermediate mesh; lower: final mesh).

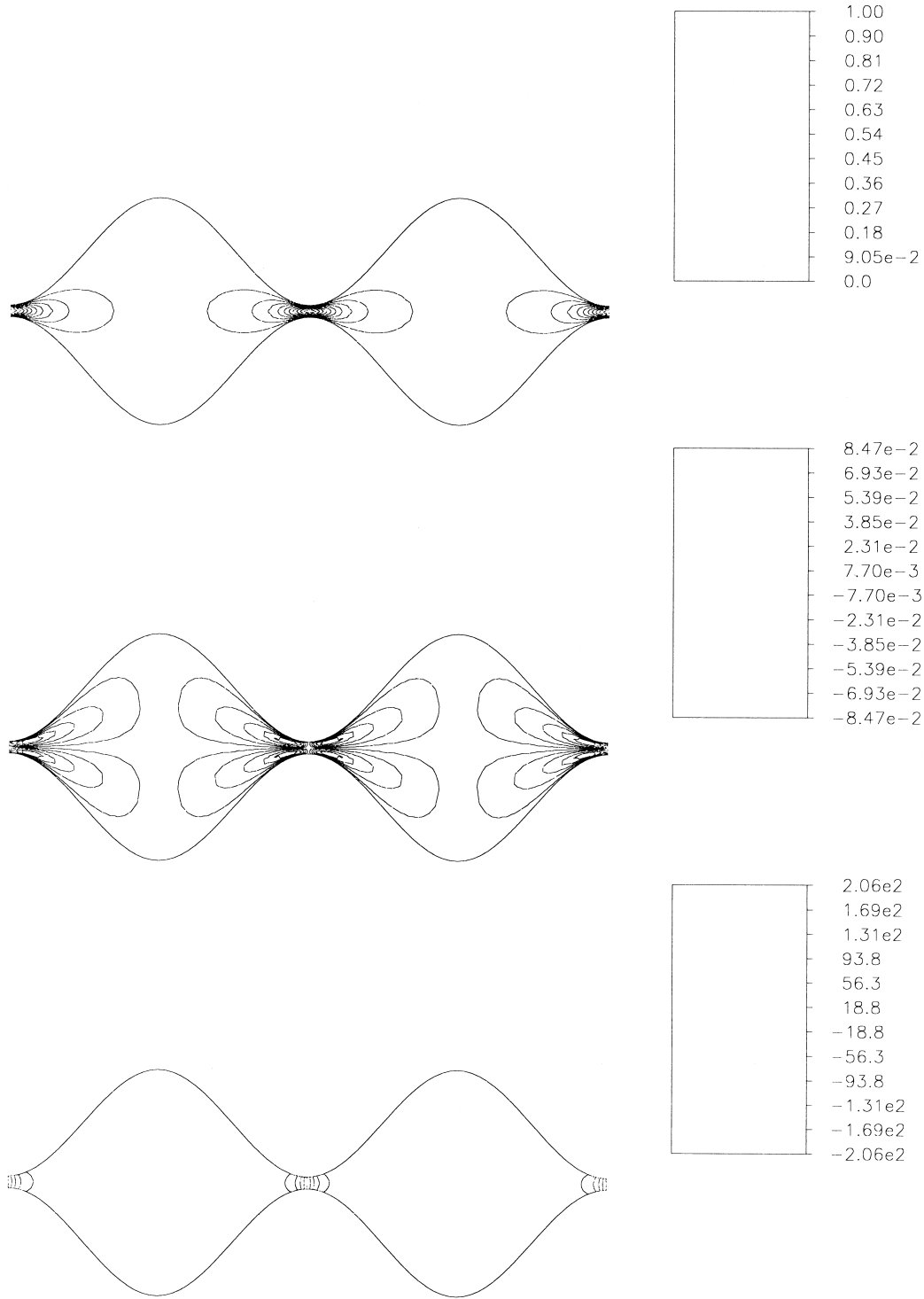
## 6 CONCLUSION AND EXTENSION

Flow in the interstices of a porous medium is typically slow, characterized by a low Reynolds number, and is described mathematically by the Stokes equations. Solutions to Stokes flow in domains with complex geometry can be obtained only numerically and usually at a high computational cost. However, when the flow domain is restricted to a small representative area, high accuracy solutions can be obtained at a reasonable cost.

This work has presented a finite element solver with adaptive mesh adaptation guided by *a posteriori* error analysis. The effectiveness of the solver has been demonstrated through numerical simulation of flow in various domains, including cases with flow separation and points

of singularity. The numerical experiments reported in this work have moderate memory and computational requirements and have all been implemented in Sun Sparc-10 workstations.

Mesh adaptation using *a posteriori* error estimates have been shown to produce accurate results, as accurate as when using a much larger number of uniform size elements. The error estimate guides us to deploy nodes and elements where they are needed and to use a coarse grid where a coarse grid is sufficient. The final grid obtained through a variety of mesh adaptation tools distributes the error of numerical approximation approximately uniformly over the flow domain. One way to judge the accuracy of the methodology is that it generally captures the physically important characteristics of the flow, such as flow separation and

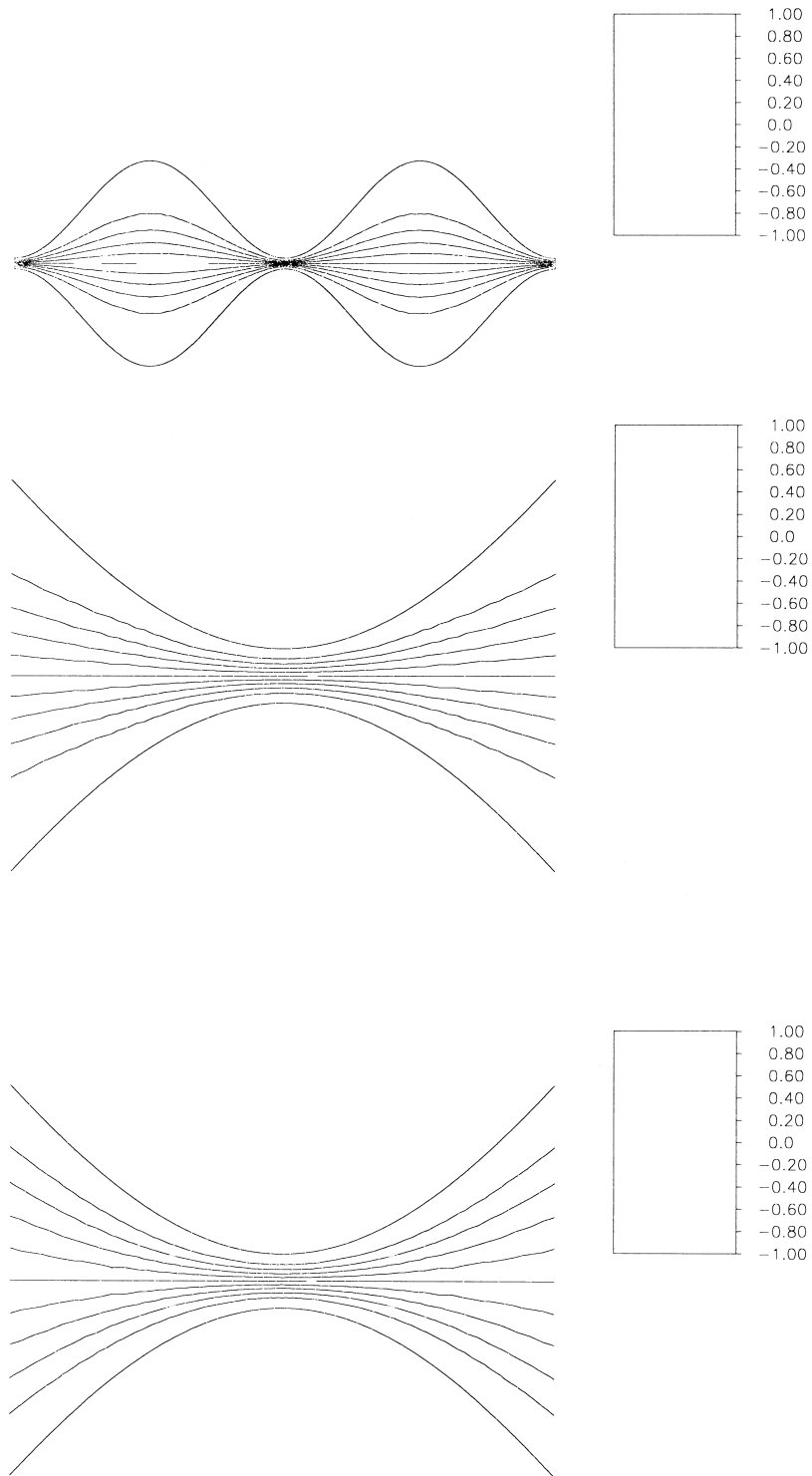


**Fig. 10.** Case 2:  $u$  (upper),  $v$  (middle), and  $p$  (lower) solutions on final mesh.

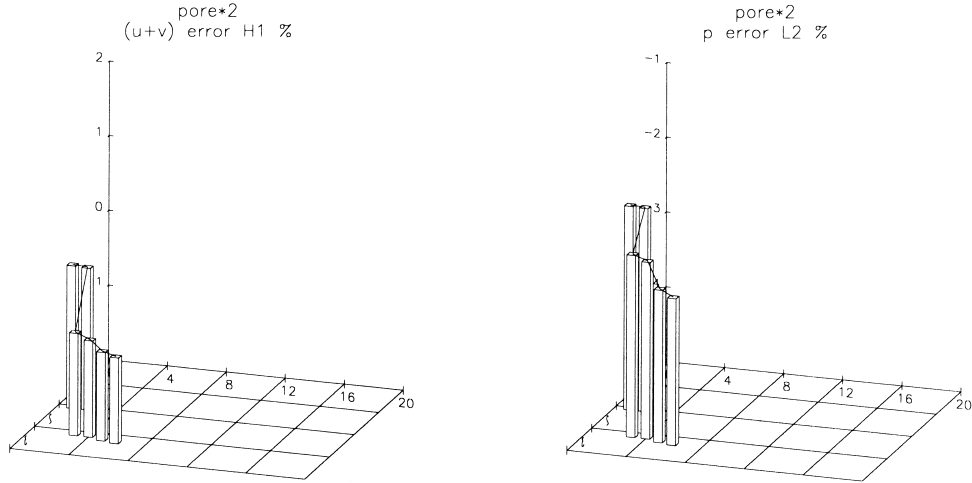
singularity points. The CPU time used for mesh adaptation is not insignificant but the results in many cases justify the cost. With the exception of trivial cases where a uniform grid is adequate, the required grid density is highly nonuniform. In some areas, important physical characteristics cannot be captured unless the grid is sufficiently fine. Using *a posteriori* error estimates provides a degree of assurance

that the solution is sufficiently accurate numerically, and implementation of mesh adaptation guides us to allocate grid nodes over the flow domain as needed for improved accuracy.

The analysis and numerical examples presented in this work are for two-dimensional flows. The methodology can be extended to three dimensions. The theory of *a posteriori*



**Fig. 11.** Case 2: streamline on final mesh (total view of numerical solution (upper); detailed views at the middle neck area of numerical solution (middle); analytical solution (lower)).

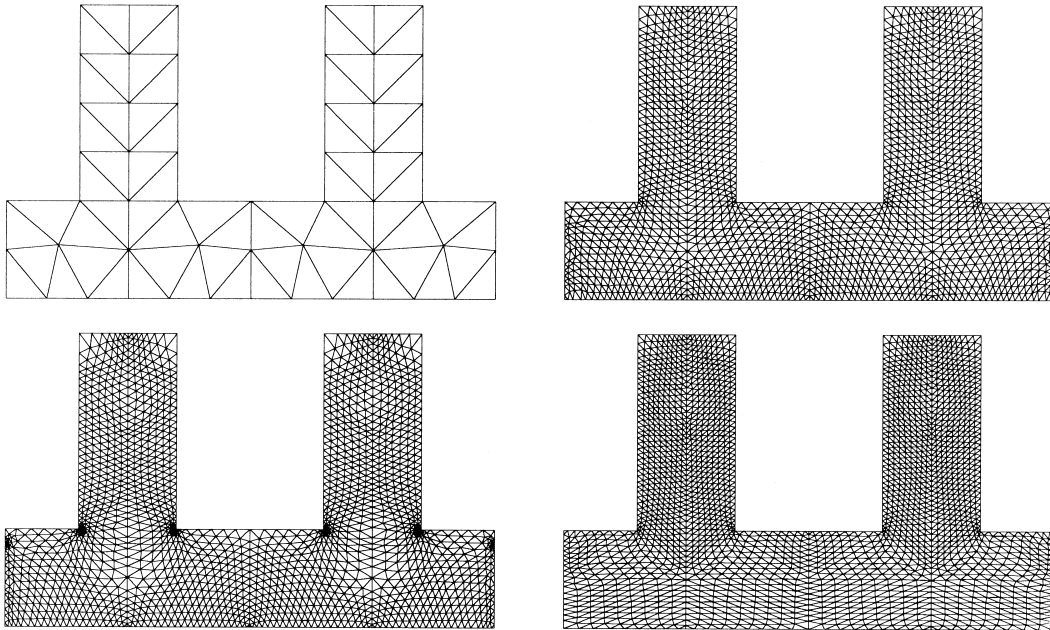


**Fig. 12.** Case 2: error evolution.

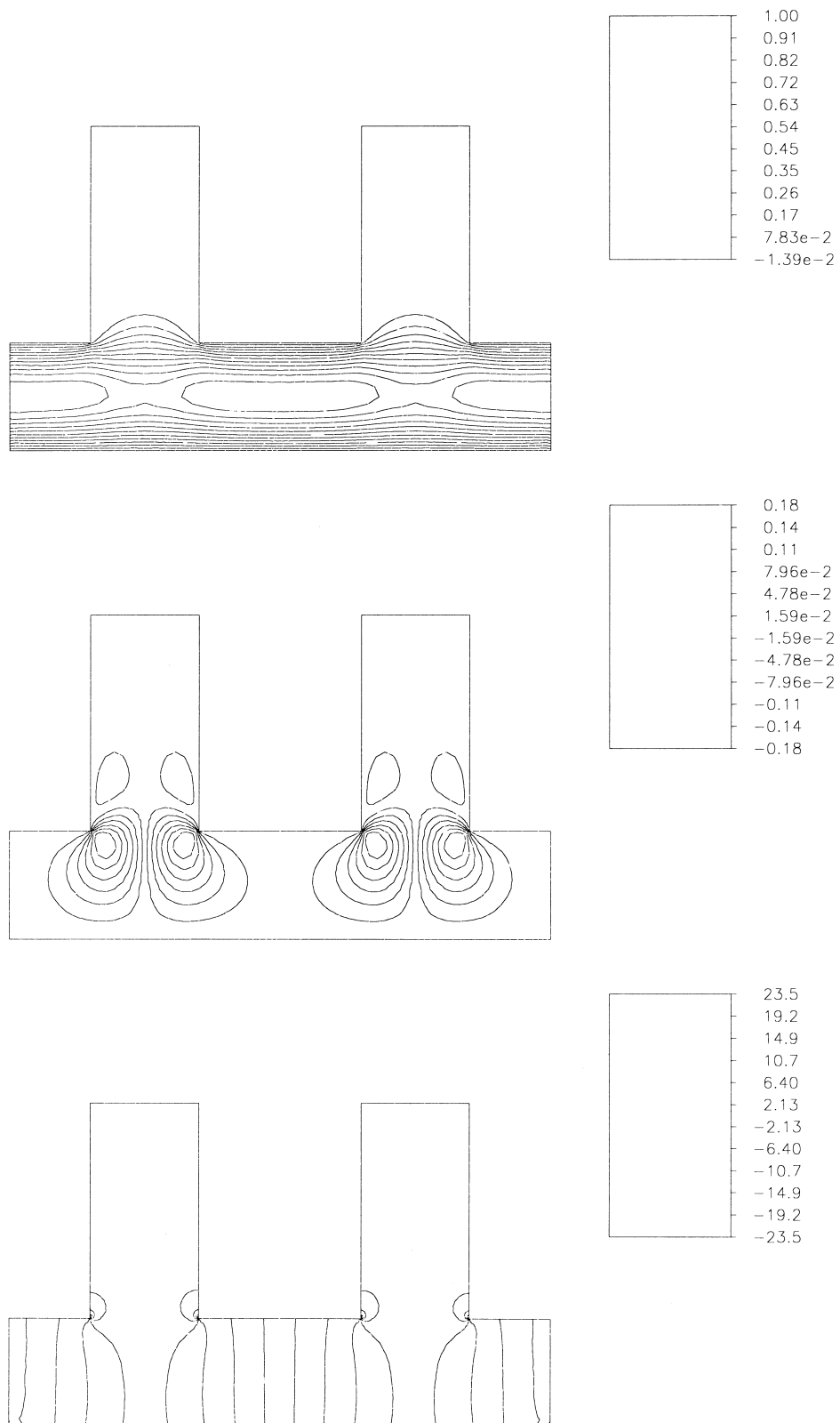
error estimation is straightforward to extend to three dimensions. The principal difficulties are algorithmic, associated with the implementation of three-dimensional finite element mesh generation and adaptation techniques. Practical algorithms are needed for refining/coarsening the grid and for moving points. In particular, it is necessary to introduce data structures recording the relationship between, for example, adjacent tetrahedrons.

## ACKNOWLEDGEMENTS

Funding for this study was provided by the National Science Foundation under project EAR-9523922 ‘Constitutive Relations of Solute Transport and Transformation’ and by the Office of Research and Development, US EPA, under agreement R-819751-01 through the Western Regional Hazardous Substance Research Center.



**Fig. 13.** Case 3: adaptive mesh evolution and another reference mesh (upper-left: initial mesh ( $NV = 55$ ,  $NT = 68$ ); upper-right: intermediate mesh ( $NV = 1807$ ,  $NT = 3332$ ); lower-left: final adaptive mesh ( $NV = 1993$ ,  $NT = 3688$ ); lower right: reference mesh ( $NV = 2251$ ,  $NT = 4212$ )).



**Fig. 14.** Case 3: solution on final mesh (upper:  $u$ ; middle:  $v$ , lower:  $p$ ).

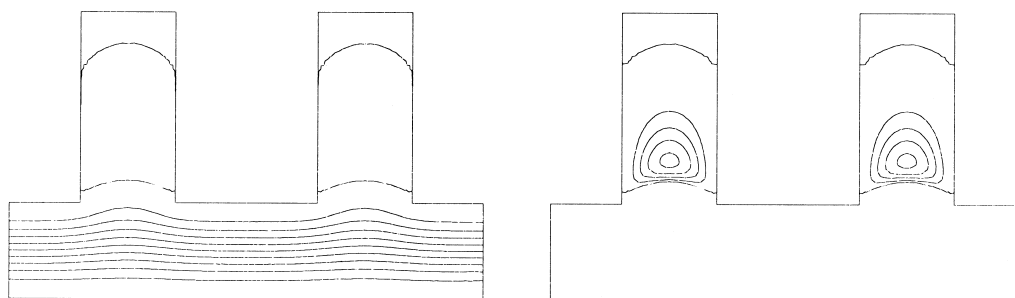


Fig. 15. Case 3: streamline on final mesh.

## REFERENCES

1. Arnold, D. N., Brezzi, F. & Fortin, F. A stable finite element for the Stokes equations. *Calcolo*, 1984, **21**(4), 337–344.
2. Babuška, I. & Rheinboldt, W. C. A posteriori error estimates for the finite element method. *Int. J. Numer. Methods Eng.*, 1978, **12**, 1597–1615.
3. Bank, R. E., Analysis of a local a posteriori error estimator for elliptic equations. In *Accuracy Estimates and Adaptivity in Finite Element Computations*, ed. I. Babuška, O. C. Zienkiewicz & E. R. A. Oliveria. Wiley, New York, 1986, pp. 119–128.
4. Bank, R. E., Dupont, T. F. & Yseretant, H. The hierarchical basis multigrid method. *Numer. Math.*, 1988, **52**, 427–458.
5. Bank, R. E., Cao, J., Mantel, B. & Périaux, J., A non-hierarchical mesh adaptive algorithm for the generalized Stokes problem. UCSD technical report, 1996.
6. Bank, R. E. & Smith, R. K. Mesh smoothing using a posteriori error estimates. *SIAM J. Numer. Anal.*, 1997, **34**, 979–997.
7. Bank, R. E. & Xu, Jinchao, The hierarchical basis multigrid method and incomplete LU decomposition. In *7th Int. Symp. on Domain Decomposition Method for Partial Differential Equations*, ed. D. Kyes & J. Xu. AMS, Providence, RI, 1994, pp.163–173.
8. Bear, J., *Dynamics of Fluids in Porous Media*. Elsevier, New York, 1972.
9. Brezzi, F. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrange multipliers. *RAIRO Anal. Numer.*, 1974, **8**, 129–151.
10. Cahouet, J. & Chabard, J. P., Multi-domains and multi-solvers finite element approach for the Stokes and shallow water equations. Rapport EDF, 41/86.03, 1986.
11. Cao, J., Estimations d'erreur a posteriori et techniques d'adaptation en éléments finis pour la simulation numérique d'écoulements de fluides visqueux. Thèse de 3e cycle, Université Pierre et Marie Curie (Paris VI), France, 1995.
12. Cao, J. & Kitanidis, P. K., Pore-scale dilution of conservative solutes: an example. *Water Resour. Res.* (accepted paper).
13. Cao, J. & Kitanidis, P. K., An algorithm for solving reactive advection-dispersion problems. *Water resour. Res.* (accepted paper).
14. Dykaar, B. B. & Kitanidis, P. K. Macrotransport of a biologically reacting solute through porous media. *Water Resour. Res.*, 1996, **32**(2), 307–320.
15. Fortin, M. & Glowinski, R., *Méthodes de Lagrangien augmenté. Méthodes mathématiques de l'informatique*, Dunos, 1982.
16. Freeze, R. A. & Cherry, J. A., *Groundwater*. Prentice Hall, 1979.
17. Frisch, U., Hasslacher, B. & Pomeau, Y. Lattice-gas automata for the Navier–Stokes equation. *Phys. Rev. Lett.*, 1986, **56**, 1505–1507.
18. Glowinski, R., *Numerical Methods for Nonlinear Variational Problems*. Springer-Verlag, New York, 1984.
19. Happel, J. & Brenner, H., *Low-Reynolds Number Hydrodynamics*. Kluwer, Boston, 1983.
20. Hasimoto, H. & Sano, O. Stokeslets and eddies in creeping flow. *Ann. Rev. Fluid Mech.*, 1980, **12**, 335–363.
21. Hughes, T. J. R., *The Finite Element Method*. Prentice Hall, 1987.
22. Kitanidis, P. K. & Dykaar, B. B. Stokes flow in a slowly varying two-dimensional periodic pore. *Transport in Porous Media*, 1997, **26**(1), 89–98.
23. Girault, V. & Raviart, P. A., *Finite Element Approximation of the Navier–Stokes Equations*. Springer Series SCM Vol. 5, 1986.
24. Labadie, G. & Lasbleiz, P., Quelques méthodes de résolution du problème de Stokes en éléments finis. Rapport EDF, HE/41/83.01, 1983.
25. Ladyzhenskaya, O. A., *The Mathematical Theory of Viscous Incompressible Flows*. Gordon and Breach, New York, 1969.
26. Lions, J. L., *Quelques Méthodes de Réolution des Problèmes aux Limites Non-linéaires*. Dunod, Paris, 1969.
27. Maitre, J. F. & Wabo, E., Stabilized formulations and mini-element for the n-dimensional Stokes equations: properties and solution procedure. In *Finite Element Methods Fifty Years of the Courant Element*, ed. M. Krizek, P. Neittaanmaki & R. Stenbert, Vol. 164 of Lecture Notes in Pure and Applied Mathematics, Marcel Dekker, 1994, pp. 285–299.
28. Pozridkidis, C., *Bounday Integral and Singularity Methods for Linearized Viscous Flow*. Cambridge University Press, 1992.
29. Rivara, M. C. Algorithms for refining triangular grids suitable for adaptive and multigrid techniques. *J. Numer. Meth. Eng.*, 1984, **20**, 745–756.
30. Spatz, W. F. & Carey, G. F., High-order compact finite difference methods with applications to viscous flows. Report 94-03 of the Texas Institute for Computational and Applied Mathematics, February 1994.
31. Teman, R., *Theory and Numerical Analysis of the Navier–Stokes Equations*. North-Holland, 1997.
32. Thomasset, F., *Implementation of Finite Element Methods for Navier–Stokes Equations*. Springer-Verlag, New York, 1981.